THE STATE OF THE ART OF COMPUTER PROGRAMMING

by

D. E. Knuth

STAN-CS-76-551
JUNE 1976

COMPUTER SCIENCE DEPARTMENT
School of Humanities and Sciences
STANFORD UNIVERSITY

The State of The Art of Computer Programming

Donald E. Knuth
Computer Science Department
Stanford University
Stanford, California 94305

This report lists all corrections and changes to volumes 1 and 3 of The Art of Computer Programming, as of May 14, 1976. The changes apply to the most recent printings of both volumes (February and March, 1975); if you have an earlier printing there have been many other changes not indicated here. Volume 2 has been completely rewritten and its second edition will be published early in 1977. For a summary of the changes made to volume 2, see SIGSAM Bulletin 9, 4 (November 1975), p. 10f -- the changes are too numerous to list except in the forthcoming book itself.

On any given day the author likes to feel that the last bug has finally disappeared, yet it appears likely that further amendments will be made as time goes by. Therefore a family of computer programs has been written to maintain a collection of errata, in the form printed here, but encoded as an ad-hoc sequence of ASCII characters. The author wishes to thank Juan Ludlow-Saldivar for the enormous amount of help he provided in order to get this system rolling. (Some readers who have access to the Stanford A.I.-Lab computer may wish to consult the change file before they report a "new" error; the file name is ACP.MAS [ART,DEK]. Entries for page nnn of volume k begin with βkOlnnn (but change the Ol to OO if nnn is the Arabic equivalent of a Roman numeral); since "β" is the control character "↑C", you may rather search for simply the string "kOlnnn". The text of the correction usually includes special codes following the symbol "|", for things like font changes, etc.)

The author thanks all the bounty hunters who have reported difficulties they spotted. The reward to first finder of each error is still $1 for the first edition and $2 for the second, gratefully paid. Volume 4 remains rather far from completion, so there is plenty of time to work all the exercises in volumes 1-3 and to catch all the remaining errors therein.

This research was supported in part by National Science Foundation grant MCS 72-03752 A03 and by the Office of Naval Research contract N00014-76-C-0330.

The Art of Computer Programming
Errata et Addenda May 14 1976

2.XVII line 5 *1*

forcing himself ⇝ being encouraged

2.XIX line 10 *2*

answer ⇝ answers

2.XIX new quote for bottom of page *3*

We can face our problem.
We can arrange such facts as we have
with order and method.
--HERCULE POIROT, in *Murder on the Orient Express* (1934)

2.4 line 23 *4*

E0. ⇝ **E0.** **(boldface)**

2.26 line 3 *5*

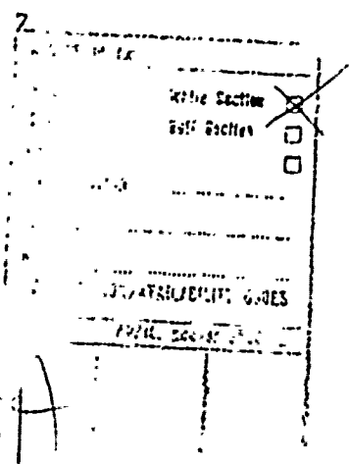
prove *A6* ⇝ prove that *A6*

2.29 line -1 *6*

$3n_0$ ⇝ $3n$

2.29 lines -3 and -2 *7*

$T < 3n_0$, where n_0 is the original value of n, ⇝ $T \leq n$,



1.26 ex 25 *8*

delete step L5 and move the ∎ to the end of step L4

1.26 ex 25, change step L3 to: *9*

L3. [Shift.] If $x-z < 1$, set $z \leftarrow z$ shifted right 1, $k \leftarrow k+1$, and repeat this step.

1.26 line 15, new sentence *10*

hardware. ⇝ hardware. The idea goes back in essence to Henry Briggs, who used it (in decimal rather than binary form) to compute logarithm tables, published in 1624.

1.27 line 23 *11*

example, ⇝ example

1.36 exercise 40 *12*

a period (.) should appear after the displayed equation

1.43 line 2 two changes *13*

(i) the (q/p) and (p/q) don't match each other. (ii) the first two lines of p44 should be moved back to p43, otherwise the reader will think exercise 47 is complete without turning the page.

1.46 line 20 *14*

$1/12n$ ⇝ $1/(12n)$

1.51 ex 15 *15*

put spaces in the first matrix, i.e.

abc ⇝ $a\ b\ c$

def ⇝ $d\ e\ f$

ghi ⇝ $g\ h\ i$

2.52 line 7 after Table 1 *16*

Shih-chieh ⇝ Shih-Chieh

2.56 left side of eq. (17) *17*

move the k a little left, to center it

2.58 line 7 after (26) *18*

Shih-chieh ⇝ Shih-Chieh

2.58 line 8 after (26) *19*

the boldface 3 appears to be in wrong font (too small)

2.72 14 places *20*

change B to B (Roman type) in the notation for Beta function, namely in line 1, line 2, line 3 (twice), line 4 (thrice), line 5 (twice), line 7, line 10 (twice), line 12, line 15.

2.72 exercise 47 *21*

in displayed formula: change upper indices from n, $n+1/2$, $2n+1$, $2n+1-k$ to r, $r-1/2$, $2r$, $2r-k$ respectively
line 3: $n = -1$. ⇝ $r = -1/2$.

2.78 lines -3 and -2 *22*

before the Renaissance. ⇝ during the Middle Ages.

2.80 line 2 *23*

1963-) ⇝ 1963-),

2.98 between (23) and (24) *24*

series ⇝ *series* (cf. (17))

1.2.2B insert new sentence just after (26): *25*

See D. A. Zave, *Inf. Proc. Letters* 5 (1976), to appear, for a further generalization.

1.2.2B replace (25) by new equation (25): *26*

$$(1/(1-z)^{m+1})\ \ln(1/(1-z)) = \Sigma_{k\geq 0}(H_{m+k}-H_m)\binom{m+k}{m} z^k,\quad m > 0.$$

1.2.5 lines 4-8 *27*

move the copy for each step to the left next to the step numbers (standard format, see e.g. Algorithm E on p2)

1.2.8 line -4 *28*

$\Sigma \rightsquigarrow \Sigma_k$

1.2.10.2 lines 3 and 4 after Fig. 11 *29*

X; that $\rightsquigarrow$ X —that
values, we $\rightsquigarrow$ values —we

1.2.11.3 line 5 *30*

distribution, the $\rightsquigarrow$ distribution, we can improve significantly on Chebyshev's inequality: The

1.2.11.3 line after (13) *31*

$f^{(2k+1)}(x)$ tends $\rightsquigarrow$ $f^{(2k+1)}(x)$ and $f^{(2k+3)}(x)$ tend

1.2.11.3 line 11 *32*

C $\rightsquigarrow$ C (Roman, not italics)

1.2.11.3 line 20 *33*

records $\rightsquigarrow$ blocks

2.155 line 5 (two places) *34*

record ⇝ block

2.156 row 5 column 4 of the table *35*

$1+T$ ⇝ $1+T$

2.165 Fig. 14 in both steps P7 and P6 *36*

PRIME [K] ⇝ PRIME[K]

2.165 line 4 *37*

fix broken type in the [of PRIME[M]

2.168 line 9 *38*

delete the exclamation point (!)

2.172 ex 3, first line of program *39*

X+1 ⇝ X+1 (0)

2.177 last line of ex 18 *40*

assume ⇝ assume that

2.184 line 5 *41*

insert more space after the period, this line's too narrow

2.191 line no. 21 of the program *42*

PERM+1 ... ⇝ PERM+1,...

2,180 line 8 *43*

itself " ⇝ itself."

2,181 top of page *44*

the "1" is broken in "1.3.3"

2,218 line 14 *45*

the 0 is broken

2,226 line 16 *46*

O. J. ⇝ O.-J.

2,237 line -10 *47*

print) ⇝ print),

2,257 Fig. 3(a) *48*

delete the funny little box which appears between "third from top" and "fourth from top"

2,258 just after (1) *49*

remove black speck

2,259 lines -3 and -2 *50*

delete the sentence "Is there ... obtainable?"

2,265 bottom line *51*

TOP ⇝ TOP (twice)

2.249 line 3 *52*

‹L‹ ⇝ ‹L‹

2.345 after step names G1 and G2 *53*

broken type ⌈ for [

2.348 line -1 *54*

BASE, BASE+1, BASE+2, ⇝ BASE+1, BASE+2, BASE+3,

2.355 in (10) *55*

move the heavy bar to the right so that it is aligned vertically with the heavy bar in (11)

2.369 comment for line 18 of the program *56*

T3 ⇝ T4

2.385 new paragraph before the exercises *57*

In spite of the fact that Algorithm T is so efficient, we will see an even better algorithm for topological sorting in Section 7.4.

2.375 changes to Program A *58*

line 04: 6H ⇝ 1H
line 05: becomes line 06
line 06: becomes line 07
line 07: becomes line 05, and delete the "1H" and change x ⇝ $1+m''$
line 12: becomes the following two lines

12	LD2	1:3(LINK)	q'	Q←LINK(Q1).
13	JMP	2B	q'	Repeat.

lines 13-35 become lines 14-36
change 6B ⇝ 1B in what was line 17 (now line 18)

2.276 line -1 59

h^3. ⇝ h^3-1.

2.276 line -4 60

exceed h ⇝ exceed h-1

2.276 line 12 61

29 ⇝ 27 (twice)

2.284 Table 1, left column 62

the line for time 0200 is out of place, it belongs just before the line for time 0256

2.286 Fig. 12 63

the shading in this figure mysteriously disappeared from the 3rd column of nodes, in the second edition. (First edition was OK!)

2.297 line 7 64

2419200 ⇝ 2,119,200

2.299 two lines before (11) 65

is the lowest value ⇝ points to the bottom-most value

2.304 exercise 20 line 3 66

A[I,J] ⇝ A[I,J]

2.304 new exercise 67

21. [20] Suggest a storage allocation function for $n \times n$ matrices where n is variable. The elements A[I,J] for $1 \le I,J \le n$ should occupy n^2 consecutive locations, regardless of the value of n.

2.322 tree illustration near bottom of page 68

the number "(9)" must be inserted at the right of this diagram

2.325 line -13 69

P* ⇝ P*

2.333 between (2) and (3) 70

tilt the diagram [illegible] and we have ⇝
tilt the diagram and bend it slightly, obtaining

2.334 Fig. 17 71

the photographer has lined up the two parts of this figure improperly in this edition; the left-hand half of the illustration should be lowered so that the trees are flush at the bottom -- this means that corresponding letters will be on the same line in both left and right parts of the illustration

2.336 line -9 72

of (7) ⇝ of the left-hand tree in (7)

2.349 line 16 73

node to ⇝ node with

2.353 line 9 74

only upward links are sufficient ⇝ upward links are sufficient by themselves

1,357 in (17) *75*

delete "." outside the boxes (for consistency in style)

1,360 exercise 11 *76*

change script $\mathcal{I}$ to italic I in five places (lines 5,6,6,23,25)

1,365 Theorem A part (a) *77*

; ⇝ .

1,375 line 4 *78*

remove hairline between "fin" and "("

1,395 line -3 *79*

or it ⇝ or

1,396 line -2 *80*

Exercise ⇝ exercise

1,405 exercise 12 *81*

Suppose ⇝ [*20*] Suppose

1,406 line -14 *82*

partic lar ⇝ particular

1,427 line -4 *83*

3). ⇝ 3).

1,459 Fig. 40 *84*

the shape of the box containing B6. should have rounded sides (like that of B2); on the other hand, the box that says "Error" should be rectangular

1,465 line -5 *85*

this displayed line should be raised half a space so that it is separated from line -4 by the same amount as it is separated from line -6

1,467 lines 4,6,7 *86*

audition ⇝ condition
emergencies ⇝ emergencies.
hence ⇝ Hence

1,470 line -7 *87*

two level ⇝ two-level

1,475 exercise 39 line 3 *88*

$N(n,m)$ ⇝ $N(n,m)/n$

1,477 line 18 *89*

2 ⇝ 2,2

1,483 first line of quote *90*

me that ⇝ *me ... that*

2.265 exercise 3 *91*

line 3: let r be ⇝ let m be
line 4: If $r = 0$, ⇝ If $m = 0$,
line 5: n/r ⇝ n/m
r and let m be ⇝ m and let n be
lines 6 and 7 (steps F4 and F5) *deleted*
line 8: **F6.** ⇝ **F4.**

2.268 better answer to exercise 3 *92*

3. -1/27, but the text hasn't defined it.

2.268 exercise 13 *93*

first sentence should become:
Add "$T < 3(n-d)+k$" to assertions *A3*, *A4*, *A5*, *A6*, where k takes the respective values 2,3,3,1.

2.268 line 16 *94*

elements a and b ⇝ elements, $a < b$,

2.271 exercise 3 *95*

the value 3 is .. *two* n^2. ⇝ $n^2 = 3$ occurs for no n, and in the second place $n^2 = 4$ occurs for *two* n.

2.271 line 10 *96*

388. ⇝ 388; V. S. Linskiĭ, *Zh. Vych. Mat. i Mat. Fiz.* **2** (1957), 90-119.

2.271 new answer replacing answer 10 *97*

9,10. No, the applications of rule (d) assume that $n \geq 0$. (The result is correct for $n = -1$ but the derivation isn't.)

2.4.7B exercise 41 line 4 *98*

1/4 ⇝ 1/8 (twice)

2.4.8S exercise 31 *99*

We have ⇝ [This sum was first obtained in closed form by J. F. Pfaff, *Nova acta acad. scient. Petr.* **11** (1797), 38-57.] We have

2.4.8B and extending to page 487 *100*

change *B* to B (Roman type) in the solutions to exercises 40, 41 (twice), 42, 48 (twice).

2.4.9E exercise 14 *101*

$n+4$ ⇝ $n+1$

2.4.9Q exercise 10 line 2 *102*

(25) ⇝ (17)

2.4.9Q exercise 15 *103*

Line 1: $zG_{n-2}(z)$, ⇝ $zG_{n-2}(z)+\delta_{n0}$,
line 3 (the displayed formula): delete the period, then add a new line:
when $z \neq -1/4$; $G_n(-1/4) = (n+1)/2^n$ for $n \geq 0$.

2.4.9B bottom of page, a new answer to exercise 1.2.11.2-3: *104*

3. $|R_{2k}| < |B_{2k}|/(2k)!\int_1^n |f^{(2k)}(x)|dx$. [C. H. Reinsch observes that $R_{2k} = \int_1^n (B_{2k+2} - B_{2k+2}(\{x\}))\, f^{(2k+2)}(x)dx/(2k+2)!$, and that $B_{2k+2} - B_{2k+2}(\{x\})$ always lies between 0 and $(2-2^{-2k-1})B_{2k+2}$. Therefore if $f^{(2k+1)}(x)$ but not $f^{(2k+3)}(x)$ tends monotonically to zero, (13) still holds for some θ with $0 < \theta < 2 - 2^{-2k-1}$.]

2.5.1E exercise 6 *105*

$O(n^{-3})$ ⇝ $O(n^{-3})$)

1.5[illegible] exercise 14 *106*

line 3: MOVE ⇝ MOVE
line 4: JSJ*+1 ⇝ JSJ *+1

1.5[illegible] exercise 17(b) *107*

(Using assembly ... section.) ⇝
(A slightly faster, but quite preposterous, program uses 993 STZ's: JMP 3995; STZ 1,2; STZ 2,2; ...; STZ 993,2; J2N 3999; DEC2 993; J2NN 3001; ENN1 0,2; JMP 3000,1.)

1.5[illegible] exercise 18 add new sentence: *108*

(Unless the program itself appears in locations 0000-0015.)

1.5[illegible] exercise 20 *109*

Fukuoka) ⇝ Fukuoka.)

1.5[illegible] exercise 16 line 1 *110*

(49): ⇝ (49);

1.5[illegible] new line just before answer no. 23: *111*

For small byte size, the entries $\pm 6^{13}$ would not appear.

1.5[illegible] exercise 6 line 3 *112*

$\sqrt{n}$ ⇝ $\sqrt{N}$

1.5[illegible] line -13 *113*

e.g. the ⇝ e.g., the

2,519 exercise 22(d) *114*

Since the α's are independently chosen, the ⇝ The

2,520 exercise 23 *115*

line 1: $\int_0^1 \ldots (\ln t)$. ⇝ $\int_0^\infty \exp(-t-E_1(t))dt$, where $E_1(x) = \int_x^\infty e^{-t}dt/t$.
line 4: $\ln n \,/\, e^\gamma$ ⇝ $e^{-\gamma} \ln n$
line 6: 8310...; ⇝ 83100 83724 31796+ [*Math. Comp.* **22** (1968), 411-415];

2,520 line 6 *116*

$\operatorname{dev} \sqrt{1/m})$. ⇝ $\operatorname{dev} \sqrt{1/m})$, when $n \geq 2m$.

2,529 line 5 *117*

process would loop indefinitely; ⇝
algorithm breaks down (possibly refers to buffer while I/O is in progress);

2,533 exercise 9 *118*

in reverse, we can get the inverse ⇝
backwards, we can get the reverse of the inverse of the reverse

2,534 exercise 12 *119*

$0 < \alpha < 1$ ⇝ $|\alpha| < 1$

2,534 line 12 *120*

$r_2(z)$ ⇝ $r_2(z)x$

2,535 exercise 4(ii) should have the following answer instead: *121*

(ii) LDA X,7:7(0:2).

2.552 new answer *122*

13. D. J. Kleitman has shown that $\lim_{n\to\infty} 2^{-n} \log f(n) = \lim_{n\to\infty} 2^{-n} \log \Pi_{0\le k\le n}\binom{n}{k}$!. [To appear.]

2.552 line -5 *123*

COUNT ⇝ COUNT

2.555 and also page 544, answer to exercise 24 *124*

replace lines 85-87 of the MIX program by

```
ST6   X,1(QLINK)          QLINK[r11] ← k.
```

Then renumber lines 88-118 to 86-116.
Finally *delete* "Note: When the ... as the loop." on p. 544.

2.555 lines 11-12 change to (with same indentation): *125*

T10. If P ≠ Λ, set QLINK[SUC(P)] ← k, P ← NEXT(P), and repeat this step.

2.555 exercise 16 *126*

line 2: 29Σ ⇝ 27Σ (twice)
line 8: 6 ⇝ 4

2.559 line -4 insert new sentence (no new paragraph) *127*

[See exercise 5.2.3-29 for a faster algorithm.]

2.559 exercise 1 line 4 *128*

AVAIL ⇝ Y ← INFO(P); AVAIL

2.559 line 2 *129*

COL(P) ⇝ COL(P0)

2.3.3 change answer 18 (saving space for new answer 21): *130*

the first part up to "after [illegible]" can be shortened as follows.
18. The three pivot steps, respectively columns 3,1,2, yield respectively
( ), ( )
(use the same matrices [illegible] squeeze onto one line)

2.3.3 exercise 20 *131*

A(1,1) ⇝ A[1,1]

2.3.3 new answer *132*

21. For example, M ← max(I,J), LOC(A[I,J]) = LOC(A[1,1]) + M(M-1) + I - J. (Such formulas have been proposed independently by many people. A. I. Rosenberg and H. R. Strong have suggested the following k-dimensional generalization: LOC(A[I_1, ..., I_k]) = L_k where L_1 = LOC(A[1, ..., 1]) + I_1 - 1, $L_r = L_{r-1} + (M_r-1)^r + (M_r - I_r)(M_r^{r-1} - (M_r-1)^{r-1})$, where $M_r = \max(I_1, ..., I_r)$. {*IBM Tech. Disclosure Bull.* **14** (1972), 3026-3028.})

2.3.3 exercise 15 *133*

remove blotches in first and second lines

2.3.3 exercise 12 line 2 *134*

A[m] ⇝ A[m],

2.3.3 new answer *135*

13. (Solution by S. Araujo.) Let steps T1 through T4 be unchanged, except that a new variable Q is initialized to A in step T1; Q will point to the last node visited, if any. Step T5 becomes two steps: 'T5. [Right branch done?] If RLINK(P) = A or RLINK(P) = Q, go on to T6; otherwise set A ← P, P ← RLINK(P) and return to T2. T6. [Visit P.] "Visit" NODE(P), set Q ← P, and return to T4.' A similar proof applies.

2.3.3 line 15 *136*

LOC(T). ⇝ LOC(T).

1.566 exercise 1 line 1 *137*

consist ⇝ consists

1.567 exercise 12 line 2 *138*

INFO(P2)-1 ⇝ TREE(INFO(P2)-1)

1.575 exercise 18 line 5 *139*

preorder ⇝ postorder

1.574 exercise 7 *140*

the diagrams for Case 1 have two arrowheads in the wrong direction ... the arrows should lead *away from* α' and *towards* β' both Before and After

1.575 line -8 *141*

332 ⇝ 322

1.576 exercise 12 line 5 *142*

$a(i)$ set $a(i) \leftarrow c(i,j)$ and $b(i) \leftarrow j$; ⇝
$a(j)$ set $a(j) \leftarrow c(i,j)$ and $b(j) \leftarrow i$;

1.577 exercise 16

line 2: th stence of ⇝ tracing out
lines 4,5: we have an oriented subsubtree ⇝ the stated digraph is an oriented tree
line 5: configuration ⇝ digraph
line 6: subtree ⇝ tree

1.578 last line *144*

D. E. Knuth, ⇝ R. Dawson and I. J. Good, *Ann. Math. Stat.* **28** (1957), 946-956; D. E. Knuth,

2.5EU exercise 24 line 2 *145*

$C' \rightsquigarrow C$

2.5EU last line of exercise 23, add: *146*

[For $m = 2$ this result is due to C. Flye Sainte-Marie, *l'Intermediaire des Mathématiciens* **1** (1894), 107-110.]

2.5E2 exercise 3 line 3 *147*

upper ⇝ right

2.5E4 exercise 10 *148*

height ⇝ weight (three times)

2.5F0 second-last line before exercise 6 *149*

this line isn't right-justified, add space after the semicolon

2.5F4 bottom line *150*

exhausted. ⇝ exhausted. [See Guy L. Steele Jr., *CACM* **18** (1975), 495-508, and P. Wadler, *CACM* **19** (1976), to appear, for further information.]
[Note that there's no comma between Steele and Jr. in his name.]

2.5F4 lines 19-21 replace by *151*

Several beautiful List-copying algorithms which make substantially weaker assumptions about List representation have been devised. See D. W. Clark, *CACM* **19** (1976), to appear, and J. M. Robson, *CACM* **19** (1976), to appear.

2.5F8 line 4 *152*

miniscule ⇝ minuscule

1.603 line before exercise 34 *153*

165.] ⟿ 165. See also E. Wegbreit, *Comp. J.* **15** (1972), 204–208; D. A. Zave, *Inf. Proc. Letters* **3** (1975), 167–169.]

1.618 line −7 *154*

det (*A*) ⟿ det(*A*)

1.619 in several places *155*

change · · to . . . in the definitions of x upper k, x lower k, n factorial, and Stirling numbers of both kinds

1.621 bottom line *156*

give section reference 1.2.5 in right-hand column

1.620 definition of Beta function *157*

B ⟿ B

1.624 line −20 (the entry for 1 degree of arc) *158*

1154 ⟿ *1155*

1.625 insert new paragraph after line 7: *159*

See the answer to exercise 1.3.3–23 for the 40-digit value of another fundamental constant.

1.627L last line *160*

9 ⟿ 9.

p. 627R 161

Araujo, Saulo, 560.

p. 628R 162

Bendix G20, 120.

p. 629L 163

Briggs, Henry, 26.

p. 629L Bolzano entry 164

delete "theorem,"

p. 629R 165

Carlyie, Thomas, xvi.

p. 629R 166

Christie Mallowan, Dame Agatha Mary Clarissa (Miller), xix.

p. 629R 167

Chu Shih-Chieh, 52, 58.

p. 629R 168

Clark, Douglas Wells, 594.

p. 629R Chebyshev's inequality entry 169

add p. 102

L.621L

170

Dawson, Reed, 578.

L.622W

171

Doyle, Sir Arthur Conan, 463.

L.622W

172

Even, Shimon, 239.

L.623L

173

Flye Sainte-Marie, Camille, 580.

L.623L

174

delete Fisher, David Allen

L.623W

175

Hamlet, Prince of Denmark, 228.

L.623W entry for Good, Irving John .

176

add p. 578

L.624W line -8

177

Exercise ~~exercise~~ exercise

L.625W

178

Kleitman, Daniel J., 511.

Knopp entry *179*

add p. 491

Krogdahl entry *180*

fix broken type

line 1 *181*

20 ~ 20.

182

Lanski, V. S., 470.

183

Path length, 399–405.

184

Pfaff, Johann Friedrich, 485.

185

Philco S2000, 120.

186

Poirot, Hercule, xix.

187

RCA 601, 120.

L.6510 *188*

Rosenberg, Arnold Leonard, 556.

L.6510 Robson entry *189*

add p. 594

L.6511 *190*

Shakespeare, William, 228, 465.

L.6511 *191*

delete Shih-chich, Chu entry

L.6511 *192*

Steele Jr., Guy Lewis (=Quux), 594.

L.6512 *193*

Strong, Hovey Raymond, Jr., 556.

L.6513 *194*

Tarjan, Robert Endre, 239.

L.6514 *195*

Wadler, Philip Lee, 594

L.6514 last line *196*

delete "theorem," (saves one line)

2.654 *197*

Wegbreit, Eliot Ben, 603.

2.654L *198*

Wise, David Stephen, 434, 595.

2.654W *199*

Zave, Derek Alan, 90, 603.

2.656 (namely the endpapers of the book) *200*

delete "**Table 1**"
also make the change specified for page 136

3.V line 4 of the Preface *201*

system ⇝ systems

3.VII line 4 *202*

forcing himself ⇝ being encouraged

3.IX line 10 *203*

answer ⇝ answers

3.XII *204*

raise this illustration about 3/8 inch

3.1 making the quotation format more consistent *205*

line 5: The Prince ⇝ *The Prince*
line 10: MASON (The Case ... 1951) ⇝
MASON, in *The Case of the Angry Mourner* (1951)

3.22 new exercises *206*

•**21.** [*M25*] (C. D. Knott.) Show that the permutation $a_1 \ldots a_n$ is obtainable with a stack, in the sense of exercise 2.2.1-5 or 2.3.1-6, if and only if $C_j \leq C_{j+1}+1$ for $1 < j < n$ in the notation of exercise 7.
22. [*M28*] (C. Meyer.) When m is relatively prime to n, we know that the sequence $(m \bmod n)$ $(2m \bmod n)$ ... $((n-1)m \bmod n)$ is a permutation of $\{1,2,\ldots,n-1\}$. Show that the number of inversions of this permutation can be expressed in terms of Dedekind sums (cf. Section 3.3.3).

3.23 line -9 *207*

45885 ⇝ 45855

3.24 lines 5-8 after (38) *208*

Curiously ... situation to the ⇝ An interesting one-to-one correspondence between such permutations and binary trees, more direct than the roundabout method via Algorithm I that we have used here, has been found by D. Rotem [*Inf. Proc. Letters* **4** (1975), 58-61]; similarly there is a

3.27 insert new sentence after (53): *209*

Actually the O terms here should have an extra 9ϵ in the exponent, but our manipulations make it clear that this 9ϵ would disappear if we had carried further accuracy.

3.72 exercise 28, three changes *210*

the average is ⇝ the average l_n is
sorting" for some obscure reason, ⇝ sorting,"
$2\sqrt{n}$; ... $1.97\sqrt{n}$.) ⇝ $2\sqrt{n}$. L. A. Shepp and B. F. Logan have proved that $\liminf_{n\to\infty} l_n/\sqrt{n} > 2$ [to appear].)

3.79 figure 9 step D3 *211*

COUNT $[K_j]$ ⇝ COUNT$[K_j]$

3.107 addition to step B2 *212*

(If BOUND = 1, this means go directly to B4.)

3.107 line -5 *213*

the underline shouldn't be broken

3.108 comments for lines 14 and 15 of the program *214*

BOUND ⇝ BOUND

3.120 line 9 *215*

(December, 1974).) ⇝ (1974), 287-289.)

3.134 line 8 *216*

$\log_2$ ⇝ lg

3.135 exercise 15 line 2 *217*

subscripts on superscripts are in wrong font

3.142 line 1 *218*

items; ⇝ items,

3.146 line 3 *219*

r15 ⇝ r15

S.155 line -18 *220*

one ⇝ at least one

S.165 last line of Table 2 *221*

179 ⇝ 170

S.200 line -2 *222*

wise, oracle ⇝ dangerous, adversary

S.200 lines -13, -12, -9, -8 *223*

pronouncements ⇝ outcomes (four changes)

S.200 lines -7 thru -3 *224*

oracles ⇝ adversaries
oracle ⇝ adversary (five changes)

S.208 lines 7-23 must be replaced by new copy: *225*

Constructing lower bounds. Theorem M shows that the "information theoretic" lower bound (2) can be arbitrarily far from the true lower bound; thus the technique used to prove Theorem M gives us another way to discover lower bounds. Such a proof technique is often viewed as the creation of an *adversary*, a pernicious being who tries to make algorithms run slowly. When an algorithm for merging decides to compare $A_i : B_j$, the adversary determines the fate of the comparison so as to force the algorithm down the more difficult path. If we can invent a suitable adversary, as in the proof of Theorem M, we can ensure that every valid merging algorithm will have to make a rather large number of comparisons. (Some people have used the words 'oracle' or 'demon' instead of 'adversary'; but it is preferable to avoid such terms in this context, since 'oracles' have quite a different connotation in the theory of recursive functions, and 'demons' appear in still a different guise within languages for artificial intelligence.)

We shall make use of *constrained adversaries*, whose power is limited with regard to the outcomes of certain comparisons. A merging method which is under the influence of a constrained adversary does not know about the constraints, so it must make the necessary comparisons even though their outcomes have been predestined. For example, in our proof of Theorem M we constrained all outcomes by condition (5), yet the merging algorithm was unable to make use of this fact in order to avoid any of the comparisons.

The constraints that we shall use in the following discussion apply to the left and right ends of the files. Left constraints are symbolized by

S.209 lines 7 and 16 *226*

questions ⟿ compar .ons
be answered ⟿ result in

S.210 lines 9, 10, 18 *227*

oracle ⟿ adversary (four changes)

S.211 line 12 *228*

then we define ⟿ thus,
to be ⟿ is

S.212 line 15 *229*

our oracle ⟿ that our adversary

S.202 line 18 *230*

the oracle ⇝ he

S.202 lines 2, 11, 16, 20, -9, -6 *231*

oracle ⇝ adversary (six changes)

S.203 line 1 *232*

ORACLE ⇝ ADVERSARY

S.204 line 4 *233*

its ⇝ his

S.207 exercise 10, line 2 *234*

oracle ⇝ adversary

S.209 exercise 23, line 6 *235*

oracle ⇝ adversary

S.210 line 2 *236*

oracle is asked ⇝ adversary is about to decide

S.210 line 3 *237*

The oracle ⇝ He.

S.210 lines 5, 11, 20, 24, 27, 30 *238*

Say ⇝ Decide (six changes)

5.3.2 line -2 *239*

"oracle", ⇝ "adversary" as in Section 5.3.2,

5.3.2 *240*

line 13: finding an oracle ⇝ constructing an adversary
line 15: oracle declare ⇝ adversary cause
lines 17, 20, 23: oracle ⇝ adversary

5.3.2 replace the eight lines preceding Table 1 by: *241*

may be subject to further improvement. The fact that $V_4(7) = 10$ shows that (11) is already off by 2 when $n = 7$.

A fairly good lower bound for the selection problem has been obtained by David G. Kirkpatrick [Ph.D. thesis, U. of Toronto, 1974], who constructed an adversary which proves that

$V_t(n) \ge n + t - 3 + \Sigma_{0<j<t-2} \lceil \lg((n+2-j)/(t+j)) \rceil$, $n \ge 2t-1$. (12)

Kirkpatrick has also established the exact behavior when $t=3$ by showing that $V_3(n) = n + \lceil \lg((n-1)/2.5) \rceil + \lceil \lg((n-1)/4) \rceil$ for all $n \ge 50$ (cf. exercise 22).

5.3.2 *242*

line 17: A. Schönhage ⇝ M. Paterson, N. Pippenger, and A. Schönhage
line 18: has ⇝ have
line -1: (12) ⇝ (13)

5.3.2 *243*

line -7: (13) ⇝ (14)
line -5: $V_t(n)$ ⇝ $\mathcal{V}_t(n)$

5.3.2 *244*

line -21: a *homogeneous* ⇝ an *oblivious*
line -2 and -1: a homogeneous ⇝ an oblivious
any homogeneous ⇝ any oblivious

S.220 lines 5-6 *245*

a suitable oracle.] ⇝ an adversary.]

S.220 substitute for exercise 22 *246*

22. [24] (David G. Kirkpatrick.) Show that when $4 \cdot 2^k < n-1 < 5 \cdot 2^k$, the upper bound (11) for $V_3(n)$ can be reduced by 1 as follows: (i) Form four "knockout trees" of size 2^k. (ii) Find the minimum of the four maxima, and discard all 2^k elements of its tree. (iii) Using the known information, build a single knockout tree of size $n-1-2^k$. (iv) Continue as in the proof of (11).

S.221 caption *247*

A homogeneous ⇝ An oblivious

S.228 line 3 *248*

1972), Chapter 15] ⇝ 1973), 163-172]

S.231 upper left corner of Fig. 51 *249*

there's a dot missing on the second line of the diagram for $n=6$

S.232 line 3 new sentence *250*

A. C. Yao and F. F. Yao have proved that $\hat{M}(2,n) = C(2,n) = \lceil \frac{3}{2}n \rceil$ and that $\hat{M}(m,n) \geq \frac{1}{2}n \lg(m+1)$ for $m < n$ [*JACM*, to appear].

S.259 line 12 *251*

15 is in the wrong bold-face font

S.259 line 13 *252*

RECORD (Q) ⇝ RECORD(Q)

3.265 line 10 *253*

delete "[*Hint:* ... 4.5.3L.]" since the proof of that theorem is being changed in the second edition of vol. 2

3.266 line 6 *254*

other P. ⇝ other P.]

3.274 line 15 *255*

to C5. ⇝ to C5 if $m > 0$.

3.354 lines -16 and -15 *256*

SORT10 ⇝ SORT10
SORT01 ⇝ SORT01

3.375 bottom line *257*

$\lg_2$ ⇝ lg (twice)

3.391 line -3 *258*

"Soundex" ⇝ contemporary form of the "Soundex"

3.397 *259*

line 17: formulated ⇝ popularized
lines 19-20: inversely ... Reading ⇝ approximately proportional to $1/n$. [*The Psycho-Biology of Language* (Boston, Mass.: Houghton Mifflin, 1935); *Human Behavior and the Principle of Least Effort* (Reading

3.408 extra annotation on line 08 of Program B *260*

LrA/2L ⇝ LrA/2L (rX changes too)

3.4.2.D line -7 *261*

only all ⇝ only if all

3.4.2.D line 13 *262*

between ⇝ between and outside the extreme values of the

3.4.2.E (6) *263*

$1 < j$ ⇝ $2 < j$

3.4.2.7 line -10 and also line -18 *264*

800 ⇝ 500

3.4.2.7 line 18 *265*

memory. It ⇝ memory. The difference between lg lg N and lg N is not substantial unless N is quite large, and typical files aren't sufficiently random either. Interpolation

3.4.2.7 new paragraph after line 14: *266*

Interpolation search is asymptotically superior to binary search: one step of binary search essentially replaces the amount of search, n, by $\frac{1}{2}n$, while one step of interpolation search essentially replaces n by $\sqrt{n}$ if the keys in the table are randomly distributed. Hence it can be shown that interpolation search takes about lg lg N steps on the average. (See exercise 22.)

3.4.2.7 replace lines -5 thru -11 by: *267*

```
01 13 09 34 29 08 08 53 20
01 13 14 31 52 30
01 13 43 10 48
01 13 48 10 30
01 14 04 26 40
49 12 27
49 09 07 12
48 49 41 15
48 16 22 59 25 25 55 33 20
48 36
```

3.4.2.8 bottom line *268*

was ⇝ seems to have been

3.4.2.9 — 269

line 11: 1 ⇝ 1,2
line 12: February, ⇝ February

3.4.2.19 line 2 — 270

the last part is in nearly perfect alphabetic order! ⇝
the alphabetic order in the last part is substantially better.

3.4.2.22 replace exercise 22 — 271

22. [*M43*] (A. C. Yao and F. F. Yao.) Show that an appropriate formulation of interpolation search requires asymptotically lg lg N comparisons, on the average, when applied to N independent uniform random keys that have been sorted. Furthermore, *all* search algorithms on such tables must make asymptotically lg lg N comparisons, on the average.

3.4.2.24 line -8 — 272

A). ⇝ A), since the necessary operations are trivial when ROOT = A.

3.4.2.28 line 17 — 273

Algorithm I. ⇝ Algorithm T

3.4.2.32 lines 7 and 8 — 274

clearly constructed $n+1$ different deletions; ⇝
constructed $n+1$ different deletions, one for each j;

3.4.2.39 — 275

line 6: A fairly ⇝ An even more
line -7: time. ⇝ time. In fact, M. Fredman has shown that $O(n)$ units of time suffice, if the right data structures are used [*ACM Symp. Theory of Comp.* 7 (1975), 240–244].

3.4389 and following pages *276*

in the second edition of vol. 3 I must revise the subsection about the Hu-Tucker algorithm to take account of the new Garsia-Wachs algorithm. Meanwhile I could have improved my treatment of Hu-Tucker by leaving the external nodes out of the priority queues (cf. (23) on p. 444, an unnecessarily cumbersome approach).

3.4389 replace lines 3-5 by: *277*

that the resulting maximum subtree weight, $\max(w(0,k-1),w(k,n))$, is as small as possible. This approach can also be fairly poor, because it may choose a node with very small p_k to be the root; however, Paul J. Bayer has proved that the resulting tree will always have a weighted path length near the optimum (see exercise 36).

3.4389 exercise 30 *278*

M46 ⇝ *M41*

3.4389 new version of exercise 36 *279*

36. [*M40*] (Paul J. Bayer.) Generalizing the upper bound of Theorem G, prove that the cost of any optimum binary search tree with nonnegative weights must be at most the total weight $S = \sum_{1\le i\le n} p_i + \sum_{0\le i\le n} q_i$ times $H + 2$, where

$$H = \sum_{1\le i\le n}(p_i/S)\lg(S/p_i) + \sum_{0\le i\le n}(q_i/S)\lg(S/q_i);$$

in fact, the top-down procedure which repeatedly chooses roots that minimize the maximum subtree weights will yield a binary search tree satisfying this bound. Show further that the cost of the optimum binary tree search tree is $\ge S$ times $H - \lg(2H/e)$.

3.4389 diagrams (2) *280*

put extra little vertical lines above the topmost nodes (B and X, respectively), for consistency with (1)

3.4389 line 2 *281*

K ⇝ *K*

3.4.60 replace lines -4 and -3 by: *282*

indicate that the average number of comparisons needed to insert the Nth item is approximately $1.01 \lg N + 0.1$ except when N is small.

3.4.62 bottom line of Table 1 *283*

2.8 ⇝ 2.78

3.4.62 Eq. (14) *284*

$p/(1-p) \approx 1.851.$ ⇝ $1/(1-p) \approx 2.851.$

3.4.62 line -12 *285*

$k - 1$ is $p/(1-p)$. ⇝ k is $1/(1-p)$.

3.4.63 *286*

line 1: $\lg N + 0.25$ ⇝ $1.01 \lg N + 0.1$
line 2: $11.17 \lg N + 4.8$ ⇝ $11.3 \lg N + 3$
line 6: $6.5 \lg N + 4.1$ ⇝ $6.6 \lg N + 3$

3.4.63 Figure 24 *287*

below the third node from the left, the 1 has a bar across it, making it look like a 4 by mistake

3.4.64 line -9 *288*

R(P) ⇝ RANK(P)

3.4.65 line 16 *289*

RANK(R). ⇝ RANK(R). Go to C10.

Ξ.4ΔΒ line -4 290

unpublished) ⇝ {see Aho, Hopcroft, and Ullman, *The Design and Analysis of Computer Algorithms* (Reading, Mass.: Addison-Wesley, 1974), Chapter 4}

Ξ.4ΔΒ lines 11-12 should be replaced by: 291

trees which arise when we allow the height difference of subtrees to be at most k. Such structures may be called HB[k] trees (meaning "height-balanced"), so that ordinary balanced trees represent the special case HB[1]. Empirical tests on HB[k] trees have been discussed by P. L. Karlton et al., *CACM* 19 (1976), 23-28.

Ξ.5.7Δ new exercise 292

31. [*M*] (M. L. Freiman.) Invent a representation of linear lists with the property that insertion of a new item between positions $m-1$ and m, given m, takes $O(\log m)$ units of time.

Ξ.ΔΔ5 new paragraph before the exercises 293

Andrew Yao has proved that the average number of nodes after random insertions without the overflow feature will be $N/(m \ln 2) + O(N/m^2)$, for large N and m, so the storage utilization will be approximately $\ln 2 \approx 69.3$ percent (*Acta Informatica*, to appear).

Ξ.5.ΔΔ line 11 294

long ⇝ long, but always a multiple of 5 characters,

Ξ.ΔΔ2 line -10 295

tree. ⇝ tric.

Ξ.5.ΔΔ line -4 296

HOUSE ⇝ HOUSE (twice)

3.4.2.6 line 5 *297*

the nodes of the tree ⇝ the tree is nonempty and that its nodes

3.4.2.8 line 19 *298*

follows. ⇝ follows:

3.5.1.0 exercise 4 *299*

there will be a new illustration, with positions numbered from 1 to 49 instead of 1 to 55. The respective entries will be:

---	(20)	---	WAS	THAT	(18)	OF
BE	THE	HIS	WHICH	WITH	THIS	---
(4)	ON	I	HE	A	OR	(19)
(3)	TO	HAD	---	(14)	BUT	(1)
(17)	FOR	BY	IN	FROM	AND	NOT
(1)	HER	ARE	IS	IT	AS	AT
(7)	---	HAVE	(3)	---	YOU	---

line 2: 55 ⇝ 49

lines after new illustration: 20,1,14,...,2 within ⇝ 20,19,3,14,1,17,1,7,3,20,18,4 within

3.5.1.5 line 2 *300*

that ⇝ that, if $n > 2$,

3.5.1.6 exercise 39 *301*

M47 ⇝ *M43*

3.5.1.8 line -5 add new sentence after "of M." *302*

(A precise formula is worked out in exercise 34.)

3.5.2.0 program line 13 *303*

empty ⇝ nonempty

3.5.2U *304*

delete lines 15–18 (m+1 not really needed after all)

3.5.2.2 line −1 *305*

antially ⇝ stantially

3.5.3.7 line 13 *306*

similar ⇝ weaker

3.5.3.8 three lines after (34) *307*

purposes. ⇝ purposes. In fact, Leo Guibas and Endre Szemerédi have succeeded in proving the difficult theorem that double hashing is asymptotically equivalent to uniform probing, in the limit as $M \to \infty$. [To appear.]

3.5.3.9 just after (37), insert new sentence: *308*

By convention we also set $f(0,0) = 1$.

3.5.3.37 new formula for (58) *309*

$C_N = 1 + (e^{-b}\alpha_b{}^b\alpha^b)/2b!)\ (2 + (\alpha-1)b + (\alpha^2+(\alpha-1)^2(b-1))R(\alpha,b)) + O(1/M)$.

3.5.4.2 line −2 *310*

until Morris's ... 1968, ⇝ until the late 1960's,

3.5.4.3 *311*

line 1: The only ... among ⇝ The first published appear .ce of the word seems to have been in H. Hellerman's book *Digital Computer System Principles* (New York: McGraw-Hill, 1967), p. 152; the only previous occurrence among
line 6: 1968 ⇝ 1967

Ξ.555 exercise 5 lines 3 and 4 *312*

ten or less ⇝ at most ten

Ξ.558 exercise 10 *313*

M48 ⇝ *M43*

Ξ.561 line -4 *314*

$M' > M$ ⇝ $M' > M$

Ξ.567 exercise 45 *315*

M48 ⇝ *M43*

Ξ.569 exercise 66 *316*

66. ⇝ ·66.

Ξ.572 new exercise *317*

67. [*M25*] (Andrew Yao.) Prove that all fixed-permutation single-hashing schemes in the sense of exercise 62 satisfy the inequality $C'_N \geq \frac{1}{2}(1 + 1/(1-\alpha))$. [*Hint*: Show that an unsuccessful search takes exactly k probes with probability $p_k \leq (M-N)/M$.]

Ξ.555 lines -10, -8, -6 *318*

LONGITUDE ⇝ LONGITUDE (three places)

Ξ.555 *319*

In the second edition I will be revising Section 6.5 again, deleting the material on post-office trees, paying more attention to Bentley's k-d trees, and discussing the search procedure of Burkhard and its analysis by Duhost and Trousse (cf. Stanford CS report of Sept. 1975)

3.555 line 13, add: *320*

[*CACM* **18** (1975), 509-516.]

3.555 line 8 *321*

3 (to appear) ⇝ **4** (1974), 1-10

3.561 the numbers in (5) should be respectively: *322*

.07948358; .00708659; .00067094; .00006786; .00000728; .00000082.

3.571 quotation *323*

Alice's Adventures in Wonderland ⇝ *Alice's Adventures in Wonderland*

3.576 lines 1-3 *324*

So we may ... $(p-1)/2$. ⇝
In general if f is any divisor of $p-1$ and d any divisor of $\gcd(f,n)$, we can similarly determine $(n/d) \bmod f$ by looking up the value of $b^{(p-1)/f}$ in a table of length f/d. If $p-1$ has the prime factors $q_1 < q_2 < \cdots < q_t$ and if q_t is small, we can therefore compute n rapidly by finding the digits from right to left in its mixed-radix representation, for radices $q_1, \ldots, q_t$. (This idea is due to R. L. Silver.)

3.579 exercise 6 *325*

the 1's in the exponents ride too high (twice)

3.580 exercise 13 *326*

b_{m-1}, ⇝ $b_{m-1}, b_{m+1},$

3.582 exercise 20 *327*

Zolnowsky (to appear). ⇝ Zolnowsky, *Discrete Math.* **9** (1974), 293-298.

Ξ.ΞΕΖ new answer *328*

22. $\lfloor mj/n \rfloor - \lfloor mi/n \rfloor - \lfloor m(j-i)/n \rfloor = 0$ or 1; and it is 0 iff $mj \bmod n > mi \bmod n$. Hence the number of inversions is $\sum_{0<i<j<n} (\lfloor mj/n \rfloor - \lfloor mi/n \rfloor - \lfloor m(j-i)/n \rfloor) = \sum_{0<r<n} \lfloor mr/n \rfloor (r - (k-r) - (k-r-1))$, which can be transformed to $\frac{1}{4}(n-1)(n-2) - \frac{1}{4}ns(m,n,0)$. [*J. für die reine und angew. Math.* **198** (1957), 162-166.]

Ξ.ΞΘΖ exercise 19 *329*

elete lines 3-7 of this answer.
line 8: The answer ⟿ (This formula
now add a new paragraph:

Note: A general formula for the number of ways to place the integers $\{1,2,\ldots,n\}$ into an array which is the "difference" of two tableau shapes $(n_1, \ldots, n_m) \setminus (l_1, \ldots, l_m)$, where $0 < l_i < n_i$ and $n = \sum n_i - \sum l_i$, has been found by W. Feit [*Proc. Amer. Math. Soc.* **4** (1953), 740-744]. This number is $n!$ det $(1/((n_j-j) - (l_i-i))!)$.

Ξ.ΞΣΖ line -4 *330*

$4.5N^2 + 2.5N - 6.$ ⟿ $(4.5N^2 + 2.5N - 6)u.$

Ξ.ΞΣΘ addendum to exercise 15 *331*

It is interesting to note that $G(w,z) = F(-wz,z)/F(-w,z)$, where $F(z,q) = \sum_{n\geq 0} z^n q^{n^2} / \prod_{1<k<n}(1-q^k)$ is the generating function for partitions $p_1 + \cdots + p_n$ into n parts, where $p_j \geq p_{j+1}+2$ for $1 \leq j < n$ and $p_n > 0$ (cf. exercise 5.1-16).

Ξ.ΕΒΖ exercise 31 line 03 *332*

INPUT-N,4 ⟿ -INPUT-N,4

Ξ.ΕΒΕ addition to answer 2 *333*

[Algorithm 5.2.3S does exactly $\text{xch}(\pi)$ exchanges, see exercise 5.2.3-4.]

Ξ.ΕΒΕ line 12 *334*

,to appear]. ⟿ **11** (1975), 29-35].

S.0321 bottom two (clobbered) lines should start respectively thus: *335*

43. As $a \to 0+$,
$\Gamma(1))/a \to \Gamma'(1) = -\gamma$,

S.0322 line -4 *336*

ζ is in wrong font (see line -2 for correct ζ)

S.0333 exercise 13 *337*

397-404 ⇝ 263-269

S.0333 lines -8, -7, -6, -3 *338*

oracle ⇝ adversary

S.0334 lines 2, 17 *339*

oracle ⇝ adversary

S.0335 exercise 9 *340*

comparisons.) ⇝ comparisons, yet the procedure is not optimal.)

S.0336 exercise 14 *341*

line 1: found in ⇝ found in $U_f(n)$ <
also add new sentence: (Kirkpatrick's adversary actually proves that (12) is a lower bound for $U_f(n+1) - 1$.)

S.0337 line 2 *342*

oracle ⇝ adversary

5.3.3.37 new answer *343*

22. In general when $2^r \cdot 2^k < n+2-t < (2^r+1) \cdot 2^k$ and $t < 2^r \leq 2t$, this procedure starting with $t+1$ knockout trees of size 2^k will yield $\lfloor (t-1)/2 \rfloor$ fewer comparisons than (11), since at least this many of the comparisons used to find the minimum in (ii) can be "reused" in (iii).

5.3.3.40 exercise 36 last line *344*

to appear.] ⟿ 333–339.]

5.3.3.40 insert new paragraph before line -2: *345*

G. Baudet and D. Stevenson have observed that exercises 37 and 38 combine to yield a simple sorting method with $(n \lg n)/k + O(n)$ comparison cycles on k processors: First sort k subfiles of size $\leq \lceil n/k \rceil$, then merge them in k passes using the "odd-even transposition merge" of order k. (To appear.)

5.3.3.42 exercise 2 line 4 *346*

D8 ⟿ C8

5.3.3.44 new answer *347*

10. See *Proc. ACM Symp. Theory of Computing* **6** (1974), 216–229.

5.3.3.45 exercise 3 for section 5.5, last line *348*

variables. ⟿ variables, without transforming the records in any way.

5.3.3.47 line -6 *349*

Strauss ⟿ Straus

5.3.3.75 exercise 7 line 3 *350*

80]. ⟿ 80; see also L. Guibas, *Acta Informatica* **4** (1975), 293–298.]

3.6773 *351*

line -9 (displayed nodes):
$r_1 \rightsquigarrow r_0$
$r_2 \rightsquigarrow r_1$
$s_1 \rightsquigarrow s_0$
$s_2 \rightsquigarrow s_1$
line -8:
$r_1 \rightsquigarrow r_0$
$s_1 \rightsquigarrow s_0$
$k > 1 \rightsquigarrow k > 0$
lines -6 and -5: the right subtrees of ... and the result $\rightsquigarrow$ the result

3.6774 new answer *352*

30. This has been proved by Russell Wessner [to appear].

3.6775 replace answer to 36 by: *353*

36. See *MAC Tech. Memo.* 69 (M.I.T., November 1975), 41 pp.

3.6776 exercise 19 *354*

the fourth rectangle in the left-hand figure is too short -- it should be extended so that its bottom line is at the same level as the bottom of the first and third rectangles

3.6777 answer 20, the line following the tree should become: *355*

It may be difficult to insert a new node at the extreme left of this tree.

3.6778 answer 30 line 4 *356*

left subtree of that $\rightsquigarrow$ subtree rooted at that

3.6778 new answer *357*

29. Partial solution by A. Yao: With $N > 6$ keys the lowest level will contain an average of $\frac{2}{3}(N+1)$ one-key nodes, $\frac{1}{3}(N+1)$ two-key nodes. The average total number of nodes lies between $0.70N$ and $0.79N$, for large N. [*Acta Informatica*, to appear.]

3.678 new answer 358

31. Use a nearly balanced tree, with additional upward links for the leftmost part, plus a stack of postponed balance factor adjustments along this path. (Each insertion does a bounded number of these adjustments.)

3.688 exercise 4 line 3 359

IONiC ⇝ TRASH
seven ⇝ *six*
insert new sentence on last line: [This remarkable 49-place packing is due to J. Scot Fishburn, who showed that 48 places do not suffice.]

3.688 new answer to exercise 11 (extends to p. 683) 360

11. No; eliminating a node with only one empty subtree will "forget" one bit in the keys of the nonempty subtree. To delete a node, it should be replaced by one of its *terminal* descendants, e.g., by searching to the right whenever possible.

3.688 exercise 12 361

line 3: Algorithm 6.2.2D. ⇝ the algorithm suggested in the previous answer.
last line: $\frac{3}{8}$ ⇝ $\frac{1}{4}$

3.688 exercise 34 line 1 362

$B_k 2^{j(k-1)}$ ⇝ $B_k/2^{j(k-1)}$

3.688 exercise 34, new answer to part (b) 363

(b) In the $1/(e^x-1)$ part, it suffices to consider values of j with $x < 2 \ln n$. For $1 < x < 2 \ln n$ we have $\Sigma_{1<k<n/x} (1-kx/n)^{n-1} + \Sigma_{k>1} e^{-kx} + O(x^2e^{-x}/n)$. For $x < 1$ we have $\Sigma_{0<k<n} \binom{n}{k} B_k(x/n)^k + \Sigma_{k=0} B_k x^k/k! + O(x^2/n)$.

3.688 line -9 364

$+f(n)_r$ ⇝ $+f(n)+2/n$,
$k<1$ ⇝ $k>1$

Ξ.6B7 new answer *365*

39. See Miyakawa, Yuba, Sugito, and Hoshi, *SIAM J. Computing*, to appear.

Ξ.6C2 line 12 *366*

and ⇝ with $0/0 = 1$ when $k = N = M-1$, and

Ξ.6C4 *367*

in the second edition I will revise several of these answers, using Mike Paterson's simplified new approach to such analyses

Ξ.6C5 exercise 39 *368*

line 6 (third line of displayed formulas): delete "$j>1$," (on this line only)
line 6 (fourth line of displayed formulas): $\binom{j}{2}$ ⇝ $\Sigma_{j>1}\binom{j}{2}$ (two places)

Ξ.6C6 new answer *369*

46. Yes. See L. Guibas (to appear).

Ξ.6C9 new answer *370*

67. Let $q_k = p_k p_{k+1} \cdots$; then $C_N = \Sigma_{k\ge 1} q_k$ and $q_k > \max(0, 1-(k-1)(M-N)/M)$.

Ξ.7B0 line 1 *371*

$\Sigma_t p_t P_{tr}$ ⇝ $\Sigma_t p_t P_{tr}$ minus the probability that a particular record is a "true drop", namely $\binom{N-q}{r-q} / \binom{N}{r}$, where $N = \binom{n}{k}$.

Ξ.7B2 line -20 last column *372*

1154 ⇝ *1155*

S.715 after line 7, a new paragraph: *373*

A few interesting constants without common names have arisen in connection with the analysis of sorting and searching algorithms; 40-digit values of these constants appear in the answers to exercises 5.2.3-27, 5.2.4-13, and 6.3-27.

S.716 left column *374*

det (*A*) ⇝ det(*A*)

S.716 line -2 *375*

. . ⇝ ...

S.717 definition of factorial *376*

1 · 2 · · *n* ⇝ 1 2 · ... *n*

S.717 definitions of x lower k and Stirling numbers of both kinds *377*

· ⇝ ...

S.717 line -12 *378*

5.1.3. ⇝ 5.1.3

S.721L *379*

Adversaries, 200-201, 209, 211-212, 220.

S.721L Aho entry *380*

add p. 468

Ξ.722L 381

Baudet, Gerard, 640.

Ξ.722R 382

Bayer, Paul Joseph, 439, 450.

Ξ.723R 383

Dedekind sums, 22.

Ξ.723R 384

Demons, 200.

Ξ.72ΞR 385

Feit, Walter, 592.

Ξ.725L 386

Fishburn. John Scot, 680.

Ξ.725L Fredman entry 387

add pp. 439, 471

Ξ.725R 388

Grasselı ∿ Grasselli

Ξ.725R Guibas entry 389

add pp. 528, 632, 696

Z.7125L

390

Hellerman, Herbert, 542.

Z.7125L

391

Hoshi, Mamoru, 687.

Z.7125L

392

delete the entry for "Homogeneous comparisons"

Z.7125L Hyafil entry

393

delete p. 215

Z.7125L two new entries

394

HB[k] trees, 468. Height-balanced trees, 468, *see* Balanced trees.

Z.7126L Knockout tournament entry

395

add pp. 214, 220

Z.7126L Linear list representation entry

396

468. ⇝ 471.

Z.7126L two new entries

397

Karlton, Philip Lewis, 468
Kirkpatrick, David Galer, 215, 220, 636.

Z.7126L

398

Logan, Benjamin Franklin, Jr., 594.

[illegible] 399

Meyer, Curt, 22.

[illegible] 400

Miyakawa, Masahiro, 687.

[illegible] 401

Oblivious algorithms, 220-221.

[illegible] 402

Oracles, 200, *see* Adversaries.

[illegible] Parallel computation entry 403

add p. 640

[illegible] 404

Paterson, Michael Stewart, 217.

[illegible] 405

Pippenger, Nicholas John, 217.

[illegible] line -1 406

78 ⤳ vi, 78

[illegible] 407

Rotem, Doron, 64.

S,7ΣUL *408*

Shepp, Lawrence Alan, 594.

S,7ΣUL line -7 *409*

223, ⇝ 223, 405 (exercise 22),

S,7ΣUW *410*

Silver, Roland Lazarus, 576.

S,7ΣUW Simultaneous comparisons entry *411*

add p. 640

S,7ΣΣL *412*

Stevenson, David, 640.

S,7ΣΣL new subentry under Sorting *413*

history of, 382-388, 417-418.

S,7ΣΣW *414*

Strauss ⇝ Straus

S,7ΣΣW *415*

Sugito, Yoshio, 687.

S,7ΣΣW *416*

Szemerédi, Endre, 528.

417

S.7211W

Tape searching, 400-401, 405.

418

S.7211W line 25

Slawomir ⇝ Slawomir

419

S.7221L

Treesort, see Tree selection sort, Heapsort.

420

S.7221L

Two-dimensional trees, 555, 570.

421

S.7221L Turski entry

Wladyslaw ⇝ Wladyslaw

422

S.7221L Ullman entry

add p. 468

423

S.7221L new subentry under Trie search

generalized, 565.

424

S.7221W

Wessner, Russell, 674.

425

S.7221W

Yao, Foong Frances, 232, 422.

ℬ.7ℰℰℳ Wrench entry *426*

add p. 686

ℬ.7ℰ:ℳ Yao, Andrew entry *427*

add pp. 232, 422, 479, 549, 678

ℬ.7ℰ5ℒ *428*

Yuba, Toshitsugu, 667.

ℬ.7ℰ5ℳ *429*

Zeta function, 612, 666.

ℬ.7ℰ5ℳ just before 2-3 trees entry *430*

2D trees, 555, 570.

ℬ.7ℰℴ (namely the endpapers of the book) *431*

delete "Table 1"
also change 1 to italic *1* in box number 35

ℬ.ℒ changes to MIX booklet *432*

p30, Fig. 3: Step P3 should say "500 found?"
p34, Fig. 4: third card should say L EQU 500
p43, line 1: 6667 ↝ 66667
p43, line 2: 193,331 ↝ 133,331
p44, problem 16, line 2: row...diagonal ↝ row and column
p44, problem 16, line 8: 10 ↝ 9
and change "record" to "block" everywhere in the discussion of MIX I/O operators.

p99, line 2: (4) ⇝ (3)
p111, lines 4 and 5, interchange the inside of the braces:
$(\{x\text{-}x^2,\ x\text{-}x^2\text{+}x^3\text{-}x^4,\ \ldots\},$
$\{x,\ x\text{-}x^2\text{+}x^3,\ x\text{-}x^2\text{+}x^3\text{-}x^4\text{+}x^5,\ \ldots\}).$
p117, problem 18, lines 3 and 4 should be:
X_L has a greatest element or is null if and on
X_R has a least element or is null.

BIBLIOGRAPHIC DATA SHEET	1. Report No. STAN-CS-76-551	2.	3. Recipient's Accession No.
4. Title and Subtitle THE STATE OF THE ART OF COMPUTER PROGRAMMING			5. Report Date June 1976 6.
7. Author(s) D. E. KNUTH			8. Performing Organization Rept. No. STAN-CS-76-551
9. Performing Organization Name and Address STANFORD UNIVERSITY COMPUTER SCIENCE DEPARTMENT STANFORD, CA. 94305			10. Project/Task/Work Unit No. 11. Contract/Grant No. N00014-76-C-0330
12. Sponsoring Organization Name and Address Office of Naval Research Department of the Navy Arlington, Virginia 22217			13. Type of Report & Period Covered Technical 14.

15. Supplementary Notes

16. Abstracts

This report lists all corrections and changes to volumes 1 and 3 of The Art of Computer Programming, as of May 14, 1976. The changes apply to the most recent printings of both volumes (February and March, 1975); if you have an earlier printing there have been many other changes not indicated here. Volume 2 has been completely rewritten and its second edition willbe published early in 1977. For a summary of the changes made to volume 2, see SIGSAM Bulletin 9, 4 (November 1975), p. 10f -- the changes are too numerous to list except in the forthcoming book itself.

17. Key Words and Document Analysis. 17c. Descriptors

17b. Identifiers/Open-Ended Terms

17c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
Approved for public release; distribution unlimited.	UNCLASSIFIED	60
	20. Security Class (This Page) UNCLASSIFIED	22. Price $4.50